

Autonomous and Mobile Robotics

Prof. Giuseppe Oriolo

ROS Tutorial: Posture Regulation with TIAGo

Michele Cipriano

DIPARTIMENTO DI INGEGNERIA INFORMATICA
AUTOMATICA E GESTIONALE ANTONIO RUBERTI

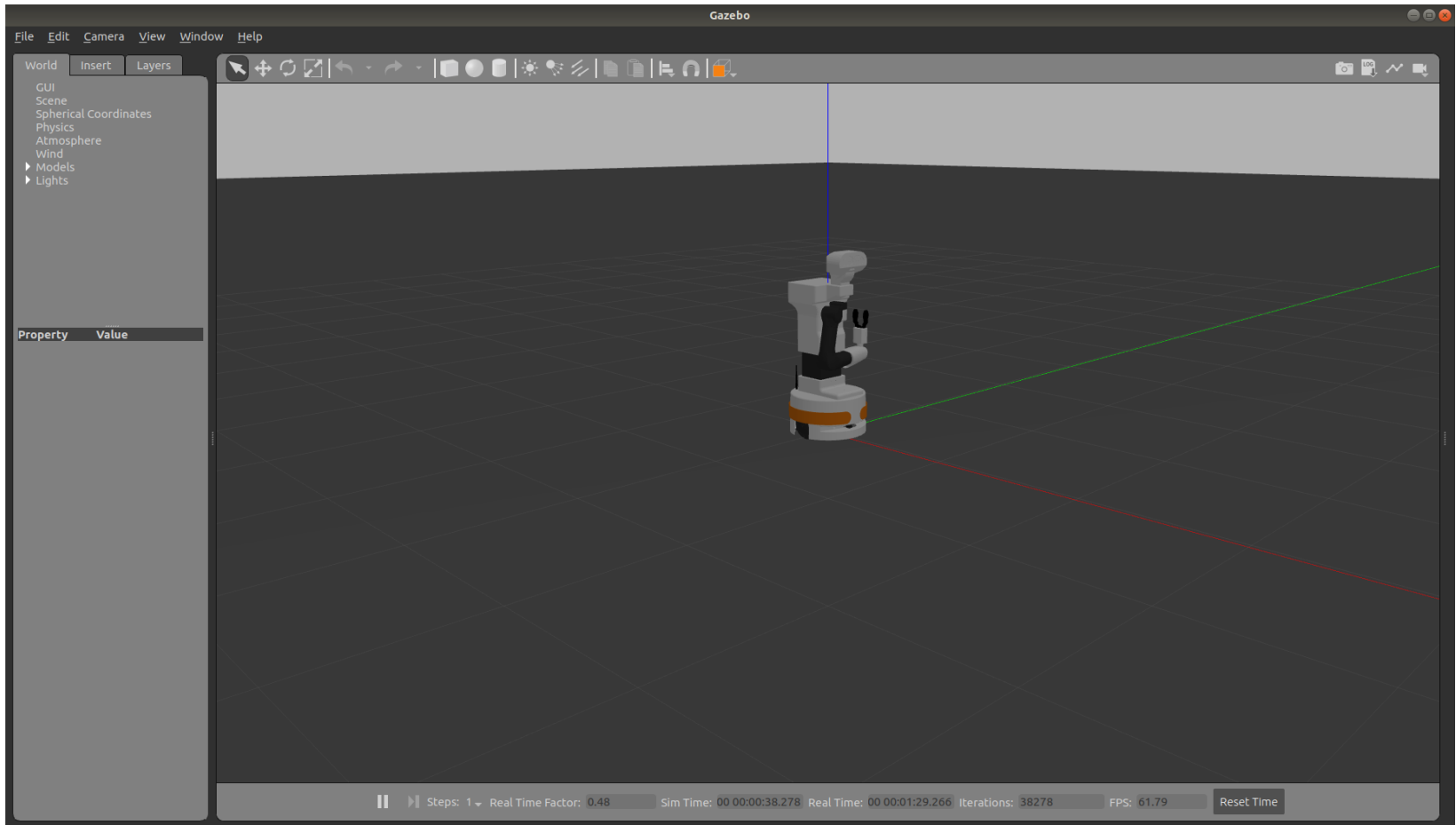


SAPIENZA
UNIVERSITÀ DI ROMA

simulating TIAGo with Gazebo

- launch Gazebo using PAL Robotics packages

```
roslaunch tiago_gazebo tiago_gazebo.launch public_sim:=true  
robot:=steel world:=empty
```



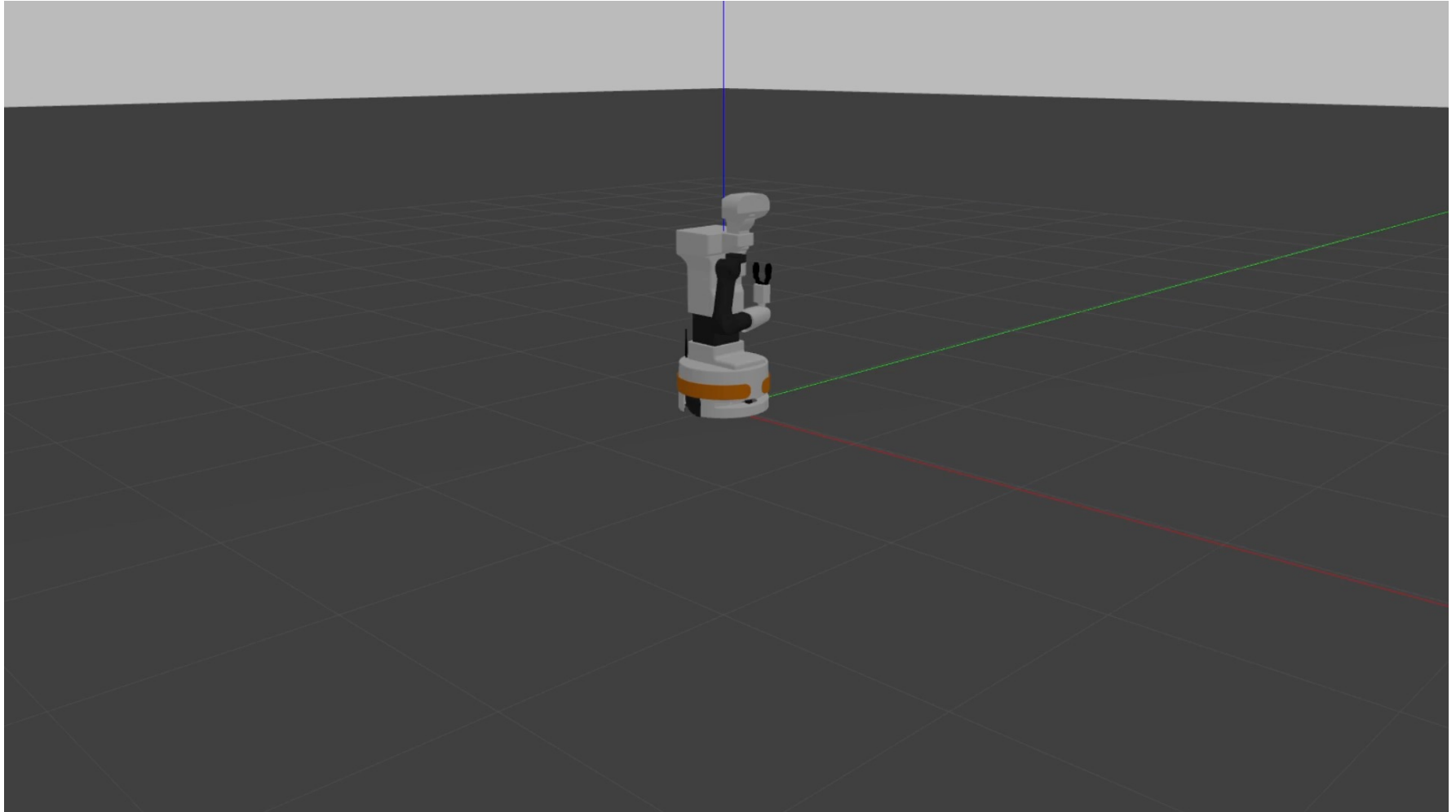
posture regulation

https://github.com/micco00x/tiago_posture_regulation

```
127
128     def start(self):
129         rate = rospy.Rate(100) # 100 Hz
130
131         # Main control loop:
132         while not rospy.is_shutdown():
133             # Read robot configuration:
134             try:
135                 transform = self.tf_buffer.lookup_transform(
136                     self.map_frame, self.base_footprint_frame, rospy.Time()
137                 )
138                 self.configuration.set_from_tf_transform(transform)
139             except (tf2_ros.LookupException,
140                   tf2_ros.ConnectivityException,
141                   tf2_ros.ExtrapolationException):
142                 rate.sleep()
143             continue
144
145         # Posture regulation:
146         command = \
147             self.posture_regulation_controller.compute_velocity_command(
148                 self.configuration,
149                 self.desired_configuration
150             )
151
152         # Send velocity commands to robot:
153         self.publish_velocity_command(command)
154
155         # Keep controller frequency at specified rate:
156         rate.sleep()
157
```

simulation

```
roslaunch tiago_posture_regulation tiago_posture_regulation.launch
```



conclusions

- ROS easily allows to implement software for robotic systems
- many open-source and ready-to-use packages for mapping, localization, planning and control, and management of specific robots
- well-written documentation and many online tutorials
- for more complex behaviors, remember that ROS1 is not real-time safe: consider using packages such as `ros_control`